

# Documentation

Student: Shangkun LI  
Student ID: 20307130125  
Department of Physics

Repository: [https://github.com/ShangkunLi/Physical\\_Design\\_Partition.git](https://github.com/ShangkunLi/Physical_Design_Partition.git)

本项目主要实现了基于FM划分算法。

## 目录

### 1 项目说明

#### 1.1 问题描述

考虑二划分问题，对给定的节点集合进行二划分，划分目标是平衡最小割。可以使用FM算法来实现。提供一个描述节点连接关系的文件prob1.txt，示例格式如下：

```
1 19
2 0 3 5 7 19
3 1 4 5 1
4 2 10 11
5 ...
```

其中第一行表示节点数，比如19。后面跟随19行，每一行的第一个数表示某一节点对应的指标，后面的数表示该节点分别连接的节点。

#### 1.2 功能说明

本项目基于FM算法实现了一个最小化割的二平衡划分算法。本项目通过定义桶形结构和节点链表等数据结构来进行具体的实现。我们设计了三种进行初始划分的方法，同时实现了单起始点多起始点方法的FM算法，并记录了最终结果。

#### 1.3 文件结构

```
1 Physical_Design_Partition # 项目文件夹
2 |— build # 编译文件夹
3 |— CMakeLists.txt # CMake文件，用于生成Makefile
4 |— include # 头文件
5 |   |— DataStructure.h
6 |   |— FMPartition.h
7 |   |— parser.h
8 |— README.md # 说明文档（本文）
9 |— src # 源文件
10 |   |— FMPartition.cpp
11 |   |— main.cpp
```

```

12 |   └─ parser.cpp
13 |   └─ tests # Benchmarks
14 |       └─ probl.txt
15 |       └─ test.txt

```

## 2 代码说明

本项目主要基于FM算法来实现最小化割的平衡二划分问题。该问题主要可分为两部分：桶形数据结构构建和节点的移动。

二分FM算法的基本思路如下：

- 对于二划分的A和B区域，首先生成一个初始划分，然后更新每个节点的增益
- 从A区域中选择增益最大的节点 $c_1$ ，将其从A区域移到B区域中，然后更新各节点增益
- 从B区域中选择增益最大的节点 $c_2$ ，将其从B区域移到A区域中，然后更新各节点增益
- 其中已经交换过的节点状态将由 `FREE` 变为 `FIXED`，直到所有节点全部变为 `FIXED` 后，一轮结束
- 记录这一轮过程中每一次 $c_1$ 和 $c_2$ 交换操作后的增益，选择增益之和最大的前 $l$ 次操作执行
- 然后进行下一轮，直到下一轮与本轮相比割不减小，停止执行，输出结果

### 2.1 桶形数据结构的构建

#### 1. 节点类

```

1  class NODE
2  {
3  public:
4      int Node_index;
5      NODE_PART Node_part;
6      STATUS Node_status;
7      int Node_gain;
8      std::vector<int> Node_connect;
9  };

```

如上述代码所示，我们定义的节点类中包括了每一个节点的指标 `Node_index`，节点当前所属的区域 `Node_part`，节点在本轮交换过程中是否已被交换 `Node_status`，节点当前的增益 `Node_gain` 还有与该节点相连的节点 `Node_connect`。在具体实现过程中，每一个节点对应一个输入的节点。

#### 2. 节点指针数组类

```

1  class NodeArray
2  {
3      std::vector<NODE *> data_array; // 存储整个图中的节点
4      NodeArray() {} // 空白初始化函数
5      NodeArray(std::vector<std::vector<int>> &graph); // 基于输入的图进行初始化
6      void copy(NodeArray na); // 将na类的所有数据复制到当前的类中，在当前类要分配相应空间
7      void reset(NodeArray na); // 将na类的所有数据复制到当前的类中，在当前类中已有相应空间
8      void recover(); // 将每个Node的状态由FIXED恢复为FREE
9      void init_half(); // 将整个NodeArray的前1000个节点划分到A区域，将NodeArray的后1000个节点划分到B区域

```

```

10 void init_even(); // 将整个NodeArray中index为奇数的Node和index为偶数的Node分别划分到
   A、B两个区域中
11 void init_rand(); // 将NodeArray中的所有Node随机均分到A、B两个区域中
12 void updateGain(); // 更新NodeArray中所有Node的增益
13 int updateGain(int i); // 更新与index为i的Node相连的所有Node的增益
14 int cutSize(); // 计算整个图划分后的割
15 };

```

在节点指针数组类中我们主要维护了一个 `std::vector<NODE *> data_array` 数据，用于快速索引图中的节点，并提供了相应的函数对图中节点进行操作。每个函数的用途详见函数声明后的注释。

### 3. 桶形数据结构类

桶形数据结构的顶层为：

```

1 class Bucket
2 {
3 public:
4     std::map<int, BucketNode *, std::greater<int>> BucketAtoB;
5     std::map<int, BucketNode *, std::greater<int>> BucketBtoA;
6     Bucket(){} // 空的初始化函数
7     void load(NodeArray &narr); // 从NodeArray中读取所有节点信息并构建桶形结构
8     int maxGain(NODE_PART partition, NodeArray &narr); // 获取某一区域内增益最大的节点
   并返回其index
9     void ChangeBetweenPart(NODE_PART partition, int i, int updateGain, int
   prevGain); // 将partition中的移动到另一个区域中去并更改相应的增益
10    void ChangeWithinPart(NODE_PART partition, int i, int updateGain, int
   prevGain); // 将每个区域内的节点移动到对应增益的BucketNode的位置
11 };

```

主要有维护了两个桶形结构 `BucketAtoB` 和 `BucketBtoA`，分别对应由A区域交换到B区域的增益桶形结构、B区域交换到A区域的增益桶形结构。桶形结构可以快速索引到每一个区域中增益最大的节点，并将其交换到另一个区域去。相关函数的具体用途详见函数声明后的注释。

`Bucket` 类的组成单元为 `BucketNode` 类。每一个 `BucketNode` 类包括 `index` 用于存储其对应的节点index，还包括指向前一个和后一个 `BucketNode` 的指针。

```

1 class BucketNode
2 {
3 public:
4     int index;
5     BucketNode *next;
6     BucketNode *prev;
7
8     BucketNode(int i, BucketNode *n, BucketNode *p)
9     {
10         index = i;
11         next = n;
12         prev = p;
13     }
14 };

```

## 2.2 节点的移动

节点的移动我们主要通过 `Partition` 类来实现。

```
1 class Partition
2 {
3 public:
4     void OneSwap(Bucket &bu, NodeArray &nodearray_local, NodeArray &nodearray_global,
5 int currentBest); // 对NodeArray完成一轮交换操作，并选择增益最大的前1次操作执行，并与已有最好的结果相比，如果割更小，则保留本轮结果
6     std::vector<std::vector<int>> FMPartition(std::vector<std::vector<int>> &graph); // 使用单个初始点完成划分算法
7     std::vector<std::vector<int>> FMPartitionRandPoint(std::vector<std::vector<int>> &graph); // 使用多个初始点选取最优结果完成划分算法
8     int mincutsizes;
9     std::vector<std::vector<int>> partition;
```

`OneSwap()` 函数利用桶形结构和节点指针数组类完成。首先从A区域中获得状态为 `FREE` 且增益最大的节点，将其移动到B区域中，并更新与该节点相连的所有节点的增益；同理，再从B区域中获得状态为 `FREE` 的节点，将其移动到A区域中，直到所有节点的状态都变为 `FIXED`，即完成一轮操作。

其中 `FMPartition()` 函数是利用单初始点的方法实现FM划分算法的，其中初始点可以用节点指针数组类中的三种方法生成，分别为 `init_half()`、`init_rand()` 和 `init_even()`。我们在 `MaxIteration` 中规定了迭代轮数的上限值，如果达到该上限值还没有达到结束条件，则强制结束，其中结束条件为连续两次迭代后割数没有减小。

但是由于FM算法是基于领域搜索的，初始点对最终结果影响很大。因此我们还用了 `FMPartitionRandPoint()` 函数来实现多起始点的FM划分算法，可以自定义起始点的个数，最后从所有起始点的迭代结果中选出最优结果作为最后的输出结果。

## 2.3 其余部分

### 1. `parser` 类

主要是读取输入文件，将文件的内容转化为后续算法处理所需要的数据结构。

注意，在我们的代码中，默认输入文件遵守如下规则：

```
1 5
2 0 2
3 1 2 3 4
4 2 1 3
5 3 1
6 4 1
```

其中第一行的5代表一共有5个节点。第二行的第一个数0代表这是index为0的节点，第二行后续数字代表与0节点相连的节点。后面四行类似。

### 2. `main.cpp`

定义了不同输入命令下执行的函数，报错信息和文件输出。

## 3 编译及结果

### 3.1 编译

从远程仓库下载源代码

```
1 | git clone https://github.com/shifengzhicheng/Routing_and_Placement.git
```

创建编译目标文件夹

```
1 | mkdir build && cd build
```

进行编译

```
1 | cmake ..  
2 | make
```

执行单初始点的FM划分算法

```
1 | ./Partition -single ../tests/probl.txt
```

若要执行多初始点的FM划分算法，则使用以下命令

```
1 | ./Partition -rand ../tests/probl.txt
```

### 3.2 结果

我们一共有两个测试文件 `test.txt` 与 `probl.txt`

1. `test.txt`

a. 执行单初始点FM划分

Terminal输出：

```
1 | iteration 0: Cut Size = 2  
2 | iteration 1: Cut Size = 2  
3 | FM Partition done. Cut Size = 2  
4 | Partition Result:  
5 | A Part:  
6 | 0 3  
7 | B Part:  
8 | 1 2 4
```

`test.txtout.txt` 输出：

```
1 partition size: 2:
2 0 3
3 partition size: 3:
4 1 2 4
5 cut size: 2
```

## b. 执行多初始点FM划分

test.txtout.txt 输出

```
1 partition size: 2:
2 0 3
3 partition size: 3:
4 1 2 4
5 cut size: 2
```

## 2. probl.txt

### a. 执行单初始点FM划分

```
1 partition size: 1000:
2 1 2 4 5 6 7 9 12 14 15 19 22 23 26 27 29 30 31 32 35 36 39 40 41 44 45 46 48 50 52
53 54 55 56 60 61 62 63 64 66 68 69 71 72 73 75 78 79 80 81 87 88 90 92 93 101 103
104 106 107 108 109 113 115 116 118 119 122 124 125 126 127 129 131 132 133 134
138 140 142 147 148 149 150 152 154 155 156 158 161 163 164 168 169 171 174 176
178 179 180 181 182 184 185 186 187 189 190 192 193 194 195 196 197 204 206 207
209 210 211 212 213 214 215 218 220 222 223 226 228 230 234 235 236 242 243 244
245 246 247 248 250 251 252 254 255 256 257 258 259 262 264 265 268 270 271 272
274 277 278 281 282 283 284 285 286 287 288 290 291 293 294 298 302 303 304 308
312 314 315 316 317 321 323 325 326 327 330 335 339 340 343 345 346 347 348 349
351 352 353 355 357 358 361 362 363 364 365 368 369 370 371 372 373 374 375 376
377 379 384 385 388 389 390 392 394 397 398 399 401 404 405 406 408 409 410 415
416 417 418 419 420 422 423 424 425 430 432 435 439 440 441 447 448 449 450 451
452 453 454 455 456 457 458 460 461 462 463 464 467 469 471 473 479 480 481 483
484 485 486 487 488 489 492 494 495 496 498 500 502 505 507 510 511 512 513 515
516 517 520 521 523 527 528 529 530 531 535 536 537 538 539 543 544 548 549 555
556 558 561 562 563 565 566 569 570 571 572 573 574 576 577 578 579 582 583 584
586 588 590 592 595 596 598 599 601 602 606 607 609 610 612 614 615 617 620 621
622 623 624 625 626 627 628 629 630 632 633 634 635 637 638 639 640 641 642 643
646 647 648 649 650 652 655 656 659 661 663 664 667 668 669 671 672 673 675 677
678 680 683 685 688 689 690 692 693 694 697 698 700 702 703 705 706 707 709 711
714 716 720 723 725 726 727 729 730 734 735 736 737 738 740 741 742 745 746 748
749 750 751 752 753 754 756 757 758 759 761 762 764 765 766 767 768 769 772 773
774 776 778 780 781 782 784 785 786 787 789 790 791 792 794 795 797 799 802 805
807 808 809 815 817 818 821 824 825 826 828 829 830 831 832 836 837 838 839 840
842 845 846 847 848 851 852 853 854 855 856 857 860 861 862 866 869 870 872 874
875 876 877 879 880 882 884 885 886 888 889 890 891 892 893 894 895 897 898 899
900 901 902 903 905 907 908 910 915 916 918 919 920 924 929 934 935 936 937 938
939 941 943 944 946 947 949 950 951 952 953 955 957 958 959 960 961 962 964 967
968 969 972 975 976 977 979 980 981 982 984 988 990 992 994 995 996 997 999 1001
1003 1005 1007 1008 1013 1016 1021 1022 1026 1027 1035 1036 1038 1039 1040 1042
```

1046 1051 1052 1053 1054 1056 1057 1061 1064 1067 1074 1075 1076 1077 1079 1080  
1081 1083 1086 1087 1088 1093 1095 1096 1102 1106 1107 1109 1110 1112 1118 1119  
1122 1125 1131 1133 1138 1139 1140 1141 1143 1146 1148 1150 1151 1153 1154 1156  
1157 1159 1164 1165 1168 1169 1171 1172 1173 1174 1180 1184 1185 1187 1190 1192  
1198 1202 1203 1205 1206 1210 1213 1214 1218 1220 1223 1226 1228 1234 1235 1236  
1239 1241 1242 1245 1246 1249 1253 1254 1258 1259 1261 1263 1265 1269 1270 1271  
1273 1274 1275 1276 1278 1282 1290 1293 1299 1300 1301 1302 1304 1305 1306 1307  
1315 1316 1325 1327 1330 1336 1339 1342 1346 1348 1349 1353 1354 1357 1359 1361  
1362 1364 1365 1366 1367 1368 1369 1372 1373 1374 1375 1376 1379 1380 1381 1386  
1388 1389 1390 1392 1394 1398 1400 1401 1403 1405 1406 1407 1410 1411 1412 1417  
1419 1420 1425 1427 1429 1433 1435 1438 1441 1443 1446 1449 1450 1451 1454 1458  
1459 1461 1462 1463 1464 1465 1467 1469 1475 1480 1482 1483 1484 1490 1491 1492  
1493 1495 1496 1497 1499 1500 1507 1508 1509 1513 1516 1517 1520 1521 1524 1525  
1528 1530 1531 1534 1537 1538 1540 1542 1543 1544 1549 1550 1552 1555 1556 1557  
1558 1560 1561 1562 1564 1565 1567 1569 1573 1575 1577 1583 1585 1587 1588 1592  
1594 1595 1597 1605 1607 1611 1612 1614 1615 1616 1617 1618 1619 1620 1624 1630  
1634 1638 1644 1645 1646 1647 1649 1651 1657 1658 1661 1662 1663 1666 1669 1670  
1674 1676 1680 1687 1688 1695 1697 1699 1700 1701 1703 1705 1706 1708 1709 1710  
1711 1713 1717 1720 1721 1724 1726 1730 1731 1733 1735 1740 1744 1745 1752 1754  
1757 1758 1759 1760 1762 1766 1767 1770 1772 1773 1776 1777 1778 1779 1782 1784  
1785 1789 1790 1792 1799 1801 1802 1810 1815 1816 1818 1823 1829 1832 1833 1834  
1835 1837 1843 1844 1847 1850 1852 1856 1863 1864 1866 1869 1870 1872 1874 1878  
1879 1883 1890 1891 1911 1916 1917 1920 1932 1933 1937 1938 1942 1944 1945 1947  
1949 1953 1954 1955 1956 1958 1959 1964 1970 1975 1976 1978 1980 1982 1983 1987  
1989 1994 1996

3 partition size: 1000:

4 0 3 8 10 11 13 16 17 18 20 21 24 25 28 33 34 37 38 42 43 47 49 51 57 58 59 65 67  
70 74 76 77 82 83 84 85 86 89 91 94 95 96 97 98 99 100 102 105 110 111 112 114 117  
120 121 123 128 130 135 136 137 139 141 143 144 145 146 151 153 157 159 160 162  
165 166 167 170 172 173 175 177 183 188 191 198 199 200 201 202 203 205 208 216  
217 219 221 224 225 227 229 231 232 233 237 238 239 240 241 249 253 260 261 263  
266 267 269 273 275 276 279 280 289 292 295 296 297 299 300 301 305 306 307 309  
310 311 313 318 319 320 322 324 328 329 331 332 333 334 336 337 338 341 342 344  
350 354 356 359 360 366 367 378 380 381 382 383 386 387 391 393 395 396 400 402  
403 407 411 412 413 414 421 426 427 428 429 431 433 434 436 437 438 442 443 444  
445 446 459 465 466 468 470 472 474 475 476 477 478 482 490 491 493 497 499 501  
503 504 506 508 509 514 518 519 522 524 525 526 532 533 534 540 541 542 545 546  
547 550 551 552 553 554 557 559 560 564 567 568 575 580 581 585 587 589 591 593  
594 597 600 603 604 605 608 611 613 616 618 619 631 636 644 645 651 653 654 657  
658 660 662 665 666 670 674 676 679 681 682 684 686 687 691 695 696 699 701 704  
708 710 712 713 715 717 718 719 721 722 724 728 731 732 733 739 743 744 747 755  
760 763 770 771 775 777 779 783 788 793 796 798 800 801 803 804 806 810 811 812  
813 814 816 819 820 822 823 827 833 834 835 841 843 844 849 850 858 859 863 864  
865 867 868 871 873 878 881 883 887 896 904 906 909 911 912 913 914 917 921 922  
923 925 926 927 928 930 931 932 933 940 942 945 948 954 956 963 965 966 970 971  
973 974 978 983 985 986 987 989 991 993 998 1000 1002 1004 1006 1009 1010 1011  
1012 1014 1015 1017 1018 1019 1020 1023 1024 1025 1028 1029 1030 1031 1032 1033  
1034 1037 1041 1043 1044 1045 1047 1048 1049 1050 1055 1058 1059 1060 1062 1063  
1065 1066 1068 1069 1070 1071 1072 1073 1078 1082 1084 1085 1089 1090 1091 1092  
1094 1097 1098 1099 1100 1101 1103 1104 1105 1108 1111 1113 1114 1115 1116 1117  
1120 1121 1123 1124 1126 1127 1128 1129 1130 1132 1134 1135 1136 1137 1142 1144  
1145 1147 1149 1152 1155 1158 1160 1161 1162 1163 1166 1167 1170 1175 1176 1177

```

1178 1179 1181 1182 1183 1186 1188 1189 1191 1193 1194 1195 1196 1197 1199 1200
1201 1204 1207 1208 1209 1211 1212 1215 1216 1217 1219 1221 1222 1224 1225 1227
1229 1230 1231 1232 1233 1237 1238 1240 1243 1244 1247 1248 1250 1251 1252 1255
1256 1257 1260 1262 1264 1266 1267 1268 1272 1277 1279 1280 1281 1283 1284 1285
1286 1287 1288 1289 1291 1292 1294 1295 1296 1297 1298 1303 1308 1309 1310 1311
1312 1313 1314 1317 1318 1319 1320 1321 1322 1323 1324 1326 1328 1329 1331 1332
1333 1334 1335 1337 1338 1340 1341 1343 1344 1345 1347 1350 1351 1352 1355 1356
1358 1360 1363 1370 1371 1377 1378 1382 1383 1384 1385 1387 1391 1393 1395 1396
1397 1399 1402 1404 1408 1409 1413 1414 1415 1416 1418 1421 1422 1423 1424 1426
1428 1430 1431 1432 1434 1436 1437 1439 1440 1442 1444 1445 1447 1448 1452 1453
1455 1456 1457 1460 1466 1468 1470 1471 1472 1473 1474 1476 1477 1478 1479 1481
1485 1486 1487 1488 1489 1494 1498 1501 1502 1503 1504 1505 1506 1510 1511 1512
1514 1515 1518 1519 1522 1523 1526 1527 1529 1532 1533 1535 1536 1539 1541 1545
1546 1547 1548 1551 1553 1554 1559 1563 1566 1568 1570 1571 1572 1574 1576 1578
1579 1580 1581 1582 1584 1586 1589 1590 1591 1593 1596 1598 1599 1600 1601 1602
1603 1604 1606 1608 1609 1610 1613 1621 1622 1623 1625 1626 1627 1628 1629 1631
1632 1633 1635 1636 1637 1639 1640 1641 1642 1643 1648 1650 1652 1653 1654 1655
1656 1659 1660 1664 1665 1667 1668 1671 1672 1673 1675 1677 1678 1679 1681 1682
1683 1684 1685 1686 1689 1690 1691 1692 1693 1694 1696 1698 1702 1704 1707 1712
1714 1715 1716 1718 1719 1722 1723 1725 1727 1728 1729 1732 1734 1736 1737 1738
1739 1741 1742 1743 1746 1747 1748 1749 1750 1751 1753 1755 1756 1761 1763 1764
1765 1768 1769 1771 1774 1775 1780 1781 1783 1786 1787 1788 1791 1793 1794 1795
1796 1797 1798 1800 1803 1804 1805 1806 1807 1808 1809 1811 1812 1813 1814 1817
1819 1820 1821 1822 1824 1825 1826 1827 1828 1830 1831 1836 1838 1839 1840 1841
1842 1845 1846 1848 1849 1851 1853 1854 1855 1857 1858 1859 1860 1861 1862 1865
1867 1868 1871 1873 1875 1876 1877 1880 1881 1882 1884 1885 1886 1887 1888 1889
1892 1893 1894 1895 1896 1897 1898 1899 1900 1901 1902 1903 1904 1905 1906 1907
1908 1909 1910 1912 1913 1914 1915 1918 1919 1921 1922 1923 1924 1925 1926 1927
1928 1929 1930 1931 1934 1935 1936 1939 1940 1941 1943 1946 1948 1950 1951 1952
1957 1960 1961 1962 1963 1965 1966 1967 1968 1969 1971 1972 1973 1974 1977 1979
1981 1984 1985 1986 1988 1990 1991 1992 1993 1995 1997 1998 1999
5 cut size: 1144

```

## b. 执行多初始点FM划分:

```

1 partition size: 1000:
2 0 1 4 7 8 9 12 13 14 18 20 21 23 25 26 28 31 32 36 39 40 41 43 45 47 48 49 50 52
56 60 61 64 65 67 69 70 74 75 76 77 78 82 84 90 92 95 96 100 101 102 105 107 108
109 110 112 114 117 119 121 123 125 126 130 131 132 134 137 138 141 144 146 147
148 149 151 155 156 158 161 162 163 164 166 167 168 169 171 173 177 178 180 182
185 189 190 192 193 197 198 200 201 203 207 209 210 212 216 218 221 222 223 225
227 228 236 239 240 242 243 244 245 247 248 251 252 255 256 257 261 263 265 266
267 270 272 276 278 279 280 281 282 284 285 287 288 289 294 297 302 303 309 310
313 315 317 318 319 321 322 323 327 328 331 332 334 335 340 344 345 346 347 348
349 350 353 355 356 359 360 362 363 364 367 368 371 374 378 379 383 384 385 387
388 389 393 396 399 402 404 405 407 408 410 411 412 413 414 416 419 421 422 423
426 435 439 440 441 442 443 446 449 450 451 457 458 459 460 461 462 464 465 467
468 469 474 476 478 479 480 481 483 484 491 494 495 499 501 502 503 505 508 510
511 513 514 515 519 522 523 525 528 529 530 533 534 535 537 538 540 543 544 548
550 552 553 557 561 563 564 566 567 568 569 570 572 575 577 578 580 581 582 583
586 587 588 589 592 593 596 598 599 602 603 605 609 610 611 612 613 615 616 617

```



619 620 621 623 624 626 629 631 632 634 635 636 639 641 643 644 645 646 648 651  
652 653 654 655 661 662 665 668 670 671 673 678 684 688 689 690 691 693 697 698  
704 706 707 708 710 712 717 719 720 722 724 725 726 727 729 730 735 739 742 743  
745 746 748 753 756 759 766 768 769 772 775 779 781 784 785 786 787 790 791 793  
794 795 796 801 802 806 807 808 809 811 813 816 819 820 821 822 829 830 833 835  
840 843 844 845 846 851 852 853 855 858 860 862 863 864 866 867 868 874 875 876  
879 880 884 886 890 894 895 897 900 903 904 905 906 907 908 910 911 921 922 924  
927 928 934 935 936 943 946 947 948 951 952 953 959 960 962 964 966 967 968 969  
970 975 977 981 983 984 986 988 992 995 997 998 999 1001 1002 1003 1004 1007 1008  
1009 1010 1011 1013 1018 1019 1022 1026 1027 1029 1030 1031 1032 1033 1034 1035  
1036 1037 1039 1042 1043 1044 1046 1047 1049 1055 1057 1058 1059 1060 1063 1074  
1080 1083 1085 1086 1089 1091 1092 1094 1095 1096 1103 1105 1107 1109 1111 1112  
1113 1115 1117 1119 1120 1121 1127 1128 1132 1133 1135 1136 1138 1139 1140 1144  
1145 1148 1149 1150 1151 1152 1153 1155 1158 1159 1160 1161 1162 1163 1164 1165  
1166 1169 1171 1173 1174 1175 1176 1178 1179 1180 1181 1182 1186 1187 1191 1193  
1203 1208 1209 1211 1213 1214 1220 1221 1224 1226 1227 1228 1229 1231 1232 1233  
1235 1237 1241 1242 1245 1246 1247 1249 1250 1251 1254 1255 1257 1266 1270 1271  
1273 1275 1277 1279 1280 1282 1283 1284 1286 1287 1289 1291 1292 1294 1295 1296  
1298 1302 1305 1309 1310 1312 1318 1320 1321 1323 1324 1328 1329 1331 1333 1338  
1341 1342 1344 1346 1348 1350 1351 1352 1354 1357 1358 1360 1363 1365 1368 1370  
1372 1374 1376 1377 1378 1380 1381 1382 1383 1387 1388 1390 1391 1392 1393 1395  
1401 1405 1406 1407 1411 1412 1413 1414 1417 1418 1419 1420 1421 1422 1424 1426  
1430 1432 1433 1437 1440 1442 1445 1446 1448 1450 1451 1453 1455 1456 1457 1458  
1459 1467 1468 1471 1474 1475 1476 1477 1478 1479 1482 1486 1490 1491 1492 1499  
1500 1501 1504 1505 1507 1511 1514 1516 1517 1518 1520 1524 1530 1532 1534 1535  
1536 1538 1539 1540 1541 1542 1544 1545 1550 1551 1552 1555 1556 1560 1564 1565  
1566 1567 1568 1569 1572 1573 1574 1576 1577 1579 1581 1582 1584 1585 1586 1587  
1588 1589 1590 1593 1595 1596 1597 1598 1600 1601 1603 1604 1605 1606 1607 1608  
1609 1611 1612 1613 1616 1617 1618 1620 1622 1623 1624 1625 1626 1627 1628 1629  
1630 1631 1633 1635 1644 1646 1652 1658 1661 1662 1663 1664 1665 1667 1669 1673  
1674 1677 1680 1682 1684 1685 1687 1691 1692 1694 1698 1699 1701 1703 1705 1706  
1707 1708 1711 1714 1715 1717 1718 1719 1720 1721 1723 1726 1728 1729 1731 1734  
1735 1736 1740 1741 1743 1744 1747 1750 1752 1753 1755 1758 1760 1762 1766 1769  
1770 1775 1776 1778 1779 1780 1782 1787 1788 1789 1790 1792 1794 1795 1796 1798  
1799 1807 1811 1812 1813 1814 1815 1816 1818 1820 1821 1823 1824 1825 1826 1830  
1832 1833 1834 1837 1838 1841 1842 1844 1845 1846 1848 1852 1854 1856 1857 1859  
1860 1862 1863 1864 1866 1868 1872 1876 1879 1880 1882 1886 1887 1889 1891 1892  
1893 1894 1895 1896 1898 1902 1903 1904 1909 1911 1914 1915 1916 1917 1919 1924  
1925 1926 1931 1932 1937 1938 1939 1940 1942 1944 1945 1947 1949 1951 1955 1956  
1957 1958 1959 1961 1962 1963 1964 1965 1967 1968 1969 1970 1972 1973 1977 1980  
1981 1982 1983 1984 1986 1987 1990 1991 1994 1997 1999

3 partition size: 1000:

4 2 3 5 6 10 11 15 16 17 19 22 24 27 29 30 33 34 35 37 38 42 44 46 51 53 54 55 57 58  
59 62 63 66 68 71 72 73 79 80 81 83 85 86 87 88 89 91 93 94 97 98 99 103 104 106  
111 113 115 116 118 120 122 124 127 128 129 133 135 136 139 140 142 143 145 150  
152 153 154 157 159 160 165 170 172 174 175 176 179 181 183 184 186 187 188 191  
194 195 196 199 202 204 205 206 208 211 213 214 215 217 219 220 224 226 229 230  
231 232 233 234 235 237 238 241 246 249 250 253 254 258 259 260 262 264 268 269  
271 273 274 275 277 283 286 290 291 292 293 295 296 298 299 300 301 304 305 306  
307 308 311 312 314 316 320 324 325 326 329 330 333 336 337 338 339 341 342 343  
351 352 354 357 358 361 365 366 369 370 372 373 375 376 377 380 381 382 386 390  
391 392 394 395 397 398 400 401 403 406 409 415 417 418 420 424 425 427 428 429

430 431 432 433 434 436 437 438 444 445 447 448 452 453 454 455 456 463 466 470  
 471 472 473 475 477 482 485 486 487 488 489 490 492 493 496 497 498 500 504 506  
 507 509 512 516 517 518 520 521 524 526 527 531 532 536 539 541 542 545 546 547  
 549 551 554 555 556 558 559 560 562 565 571 573 574 576 579 584 585 590 591 594  
 595 597 600 601 604 606 607 608 614 618 622 625 627 628 630 633 637 638 640 642  
 647 649 650 656 657 658 659 660 663 664 666 667 669 672 674 675 676 677 679 680  
 681 682 683 685 686 687 692 694 695 696 699 700 701 702 703 705 709 711 713 714  
 715 716 718 721 723 728 731 732 733 734 736 737 738 740 741 744 747 749 750 751  
 752 754 755 757 758 760 761 762 763 764 765 767 770 771 773 774 776 777 778 780  
 782 783 788 789 792 797 798 799 800 803 804 805 810 812 814 815 817 818 823 824  
 825 826 827 828 831 832 834 836 837 838 839 841 842 847 848 849 850 854 856 857  
 859 861 865 869 870 871 872 873 877 878 881 882 883 885 887 888 889 891 892 893  
 896 898 899 901 902 909 912 913 914 915 916 917 918 919 920 923 925 926 929 930  
 931 932 933 937 938 939 940 941 942 944 945 949 950 954 955 956 957 958 961 963  
 965 971 972 973 974 976 978 979 980 982 985 987 989 990 991 993 994 996 1000 1005  
 1006 1012 1014 1015 1016 1017 1020 1021 1023 1024 1025 1028 1038 1040 1041 1045  
 1048 1050 1051 1052 1053 1054 1056 1061 1062 1064 1065 1066 1067 1068 1069 1070  
 1071 1072 1073 1075 1076 1077 1078 1079 1081 1082 1084 1087 1088 1090 1093 1097  
 1098 1099 1100 1101 1102 1104 1106 1108 1110 1114 1116 1118 1122 1123 1124 1125  
 1126 1129 1130 1131 1134 1137 1141 1142 1143 1146 1147 1154 1156 1157 1167 1168  
 1170 1172 1177 1183 1184 1185 1188 1189 1190 1192 1194 1195 1196 1197 1198 1199  
 1200 1201 1202 1204 1205 1206 1207 1210 1212 1215 1216 1217 1218 1219 1222 1223  
 1225 1230 1234 1236 1238 1239 1240 1243 1244 1248 1252 1253 1256 1258 1259 1260  
 1261 1262 1263 1264 1265 1267 1268 1269 1272 1274 1276 1278 1281 1285 1288 1290  
 1293 1297 1299 1300 1301 1303 1304 1306 1307 1308 1311 1313 1314 1315 1316 1317  
 1319 1322 1325 1326 1327 1330 1332 1334 1335 1336 1337 1339 1340 1343 1345 1347  
 1349 1353 1355 1356 1359 1361 1362 1364 1366 1367 1369 1371 1373 1375 1379 1384  
 1385 1386 1389 1394 1396 1397 1398 1399 1400 1402 1403 1404 1408 1409 1410 1415  
 1416 1423 1425 1427 1428 1429 1431 1434 1435 1436 1438 1439 1441 1443 1444 1447  
 1449 1452 1454 1460 1461 1462 1463 1464 1465 1466 1469 1470 1472 1473 1480 1481  
 1483 1484 1485 1487 1488 1489 1493 1494 1495 1496 1497 1498 1502 1503 1506 1508  
 1509 1510 1512 1513 1515 1519 1521 1522 1523 1525 1526 1527 1528 1529 1531 1533  
 1537 1543 1546 1547 1548 1549 1553 1554 1557 1558 1559 1561 1562 1563 1570 1571  
 1575 1578 1580 1583 1591 1592 1594 1599 1602 1610 1614 1615 1619 1621 1632 1634  
 1636 1637 1638 1639 1640 1641 1642 1643 1645 1647 1648 1649 1650 1651 1653 1654  
 1655 1656 1657 1659 1660 1666 1668 1670 1671 1672 1675 1676 1678 1679 1681 1683  
 1686 1688 1689 1690 1693 1695 1696 1697 1700 1702 1704 1709 1710 1712 1713 1716  
 1722 1724 1725 1727 1730 1732 1733 1737 1738 1739 1742 1745 1746 1748 1749 1751  
 1754 1756 1757 1759 1761 1763 1764 1765 1767 1768 1771 1772 1773 1774 1777 1781  
 1783 1784 1785 1786 1791 1793 1797 1800 1801 1802 1803 1804 1805 1806 1808 1809  
 1810 1817 1819 1822 1827 1828 1829 1831 1835 1836 1839 1840 1843 1847 1849 1850  
 1851 1853 1855 1858 1861 1865 1867 1869 1870 1871 1873 1874 1875 1877 1878 1881  
 1883 1884 1885 1888 1890 1897 1899 1900 1901 1905 1906 1907 1908 1910 1912 1913  
 1918 1920 1921 1922 1923 1927 1928 1929 1930 1933 1934 1935 1936 1941 1943 1946  
 1948 1950 1952 1953 1954 1960 1966 1971 1974 1975 1976 1978 1979 1985 1988 1989  
 1992 1993 1995 1996 1998  
 cut size: 1109